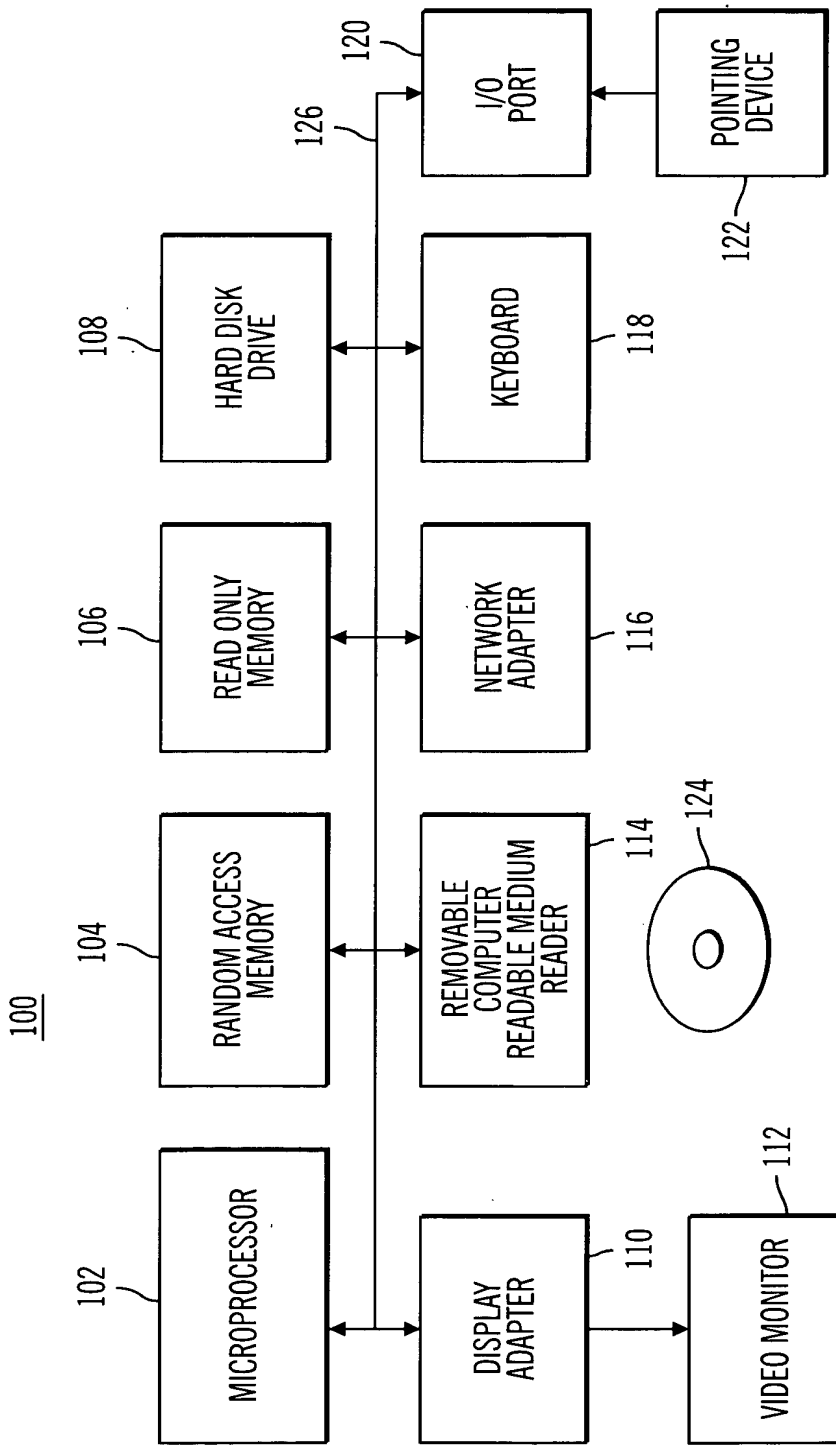

100
MICROPROCESSOR
102
RANDOM ACCESS MEMORY
104
READ ONLY MEMORY
106
HARD DISK DRIVE
108
DISPLAY ADAPTER
110
VIDEO MONITOR
112
REMOVABLE COMPUTER READABLE MEDIUM READER
114
NETWORK ADAPTER
116
KEYBOARD
118
I/O PORT
120
POINTING DEVICE
122
124
126

FIG. 1

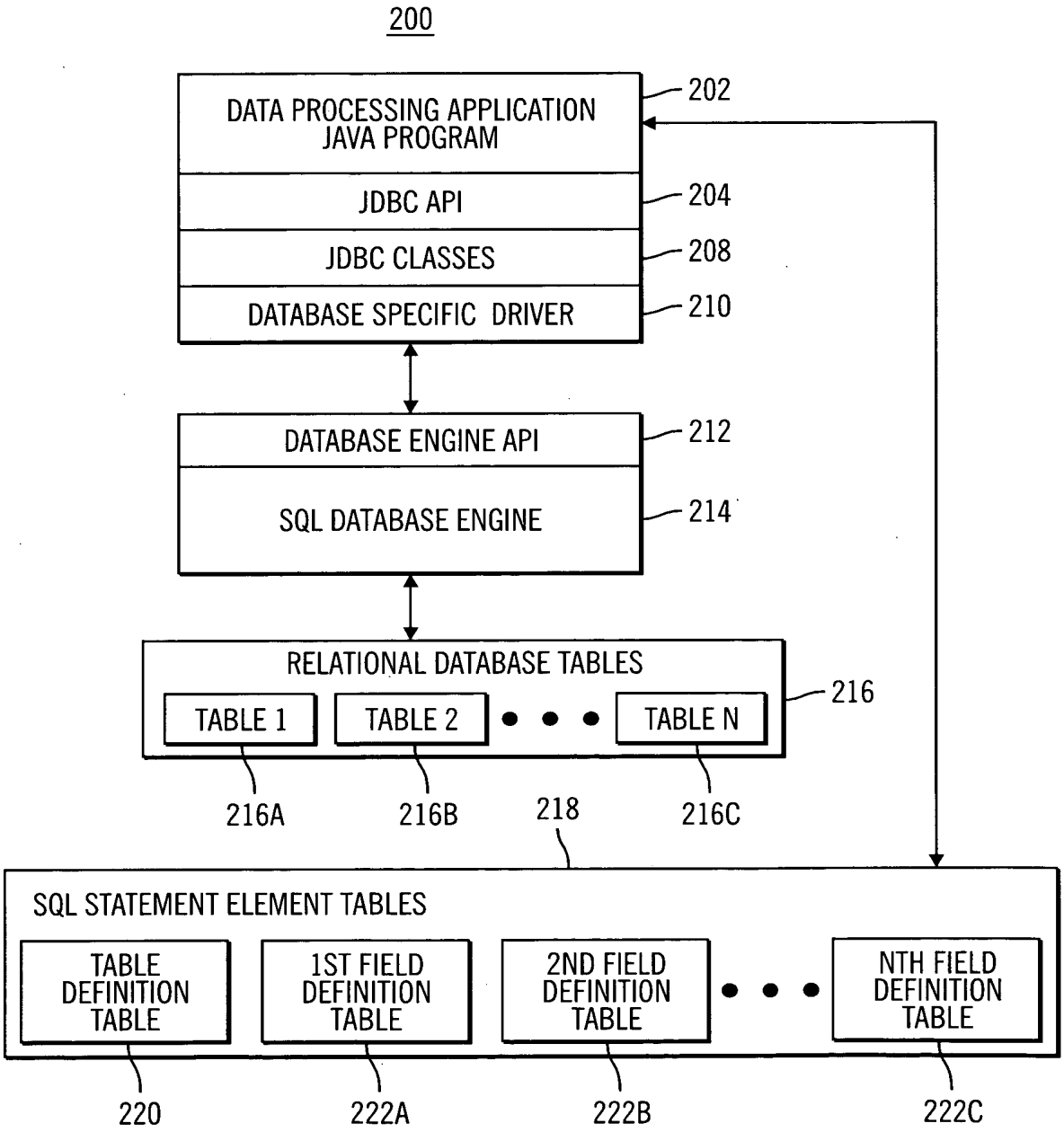

200
DATA PROCESSING APPLICATION
JAVA PROGRAM
202
JDBC API
204
JDBC CLASSES
208
DATABASE SPECIFIC DRIVER
210
DATABASE ENGINE API
212
SQL DATABASE ENGINE
214
RELATIONAL DATABASE TABLES
TABLE 1
TABLE 2
TABLE N
216
216A
216B
218
216C
SQL STATEMENT ELEMENT TABLES
TABLE DEFINITION TABLE
1ST FIELD DEFINITION TABLE
2ND FIELD DEFINITION TABLE
NTH FIELD DEFINITION TABLE
220
222A
222B
222C

FIG. 2

EXAMPLE TABLE DEFINITION TABLE

300

EXPANDS 220

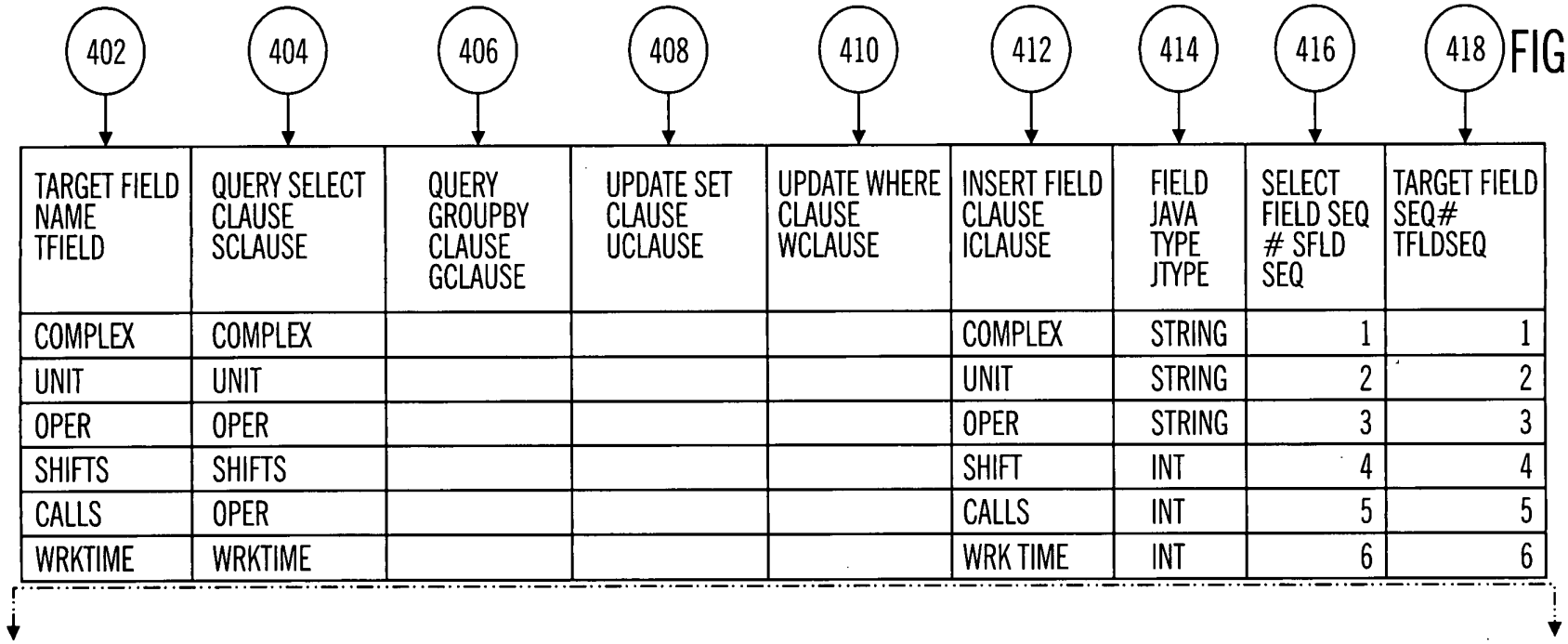
302	304	306	308	310	312
TARGET TABLENAME UTABLENAME ITABLENAME	QUERY TABLENAME QTABLENAME	QUERY SEARCH CLAUSE QSEARCH	TARGET ACTION ACTION	FIELDDEF TABLENAME FTABLENAME	TABLE SEQ # TABSEQ
OPERSTAT	TEMPSTAT		INSERT	OPERSTATTAB	1
UNITSTAT	OPERSTAT		UPDATE/INSERT	UNITSTATTAB	2
COMPLEXSTAT	UNITSTAT	COMPLEX ID=1	UPDATE	COMPLEXTAB	3
COMPLEXSTAT	UNITSTAT	COMPLEX ID=2	UPDATE	COMPLEXTAB	3

FIG. 3

EXAMPLE FIELD DEFINITION TABLES

FIELD DEFINITION TABLE: OPERSTAT

400 EXPANDS 222A



402 TARGET FIELD NAME TFIELD	404 QUERY SELECT CLAUSE SCLAUSE	406 QUERY GROUPBY CLAUSE GCLAUSE	408 UPDATE SET CLAUSE UCLAUSE	410 UPDATE WHERE CLAUSE WCLAUSE	412 INSERT FIELD CLAUSE ICLAUSE	414 FIELD JAVA TYPE JTYPE	416 SELECT FIELD SEQ # SFLD SEQ	418 TARGET FIELD SEQ# TFLDSEQ
COMPLEX	COMPLEX				COMPLEX	STRING	1	1
UNIT	UNIT				UNIT	STRING	2	2
OPER	OPER				OPER	STRING	3	3
SHIFTS	SHIFTS				SHIFT	INT	4	4
CALLS	OPER				CALLS	INT	5	5
WRKTIME	WRKTIME				WRK TIME	INT	6	6

FIG. 4A

FIELD DEFINITION TABLE: UNITSTATTAB

420 EXPANDS 222B

422 TARGET FIELD NAME TFIELD	424 QUERY SELECT CLAUSE SCLAUSE	426 QUERY GROUPBY CLAUSE GCLAUSE	428 UPDATE SET CLAUSE UCLAUSE	430 UPDATE WHERE CLAUSE WCLAUSE	432 INSERT FIELD CLAUSE ICLAUSE	434 FIELD JAVA TYPE JTYPE	436 SELECT FIELD SEQ # SFLD SEQ	438 TARGET FIELD SEQ # TFLDSEQ
COMPLEX	COMPLEX	COMPLEX		COMPLEX =?	COMPLEX	STRING	1	1
UNIT	UNIT	UNIT		UNIT =?	UNIT	STRING	2	2
CALLS	SUM(CALLS)		CALLS =?		CALLS	INT	3	3
WRK TIME	SUM(WRKTIME)		WRKTIME =?		WRKTIME	INT	4	4
CUMCALLS			CUMCALLS = CUMCALLS +?		CUMCALLS	INT	3	5
CUMWRKTIME			CUMWRKTIME = CUMWRKTIME +?		CUMWRKTIME	INT	4	6

FIG. 4B

FIELD DEFINITION TABLE: COMPLEXSTATTAB

440 EXPANDS 222C

442 TARGET FIELD NAME TFIELD	444 QUERY SELECT CLAUSE SCLAUSE	446 QUERY GROUPBY CLAUSE GCLAUSE	448 UPDATE SET CLAUSE UCLAUSE	450 UPDATE WHERE CLAUSE WCLAUSE	452 INSERT FIELD CLAUSE ICLAUSE	454 FIELD JAVA TYPE JTYPE	456 SELECT FIELD SEQ #SFLD SEQ	458 TARGET FIELD SEQ # TFLDSEQ
COMPLEX	COMPLEX			COMPLEX = ?		STRING	1	1
CALLS	SUM(CALLS)		CALLS=CALLS +?			INT	2	2
HRSWORK	SUM(WRKTIME)		HRSWORK=HRSWORK+?			INT	3	3

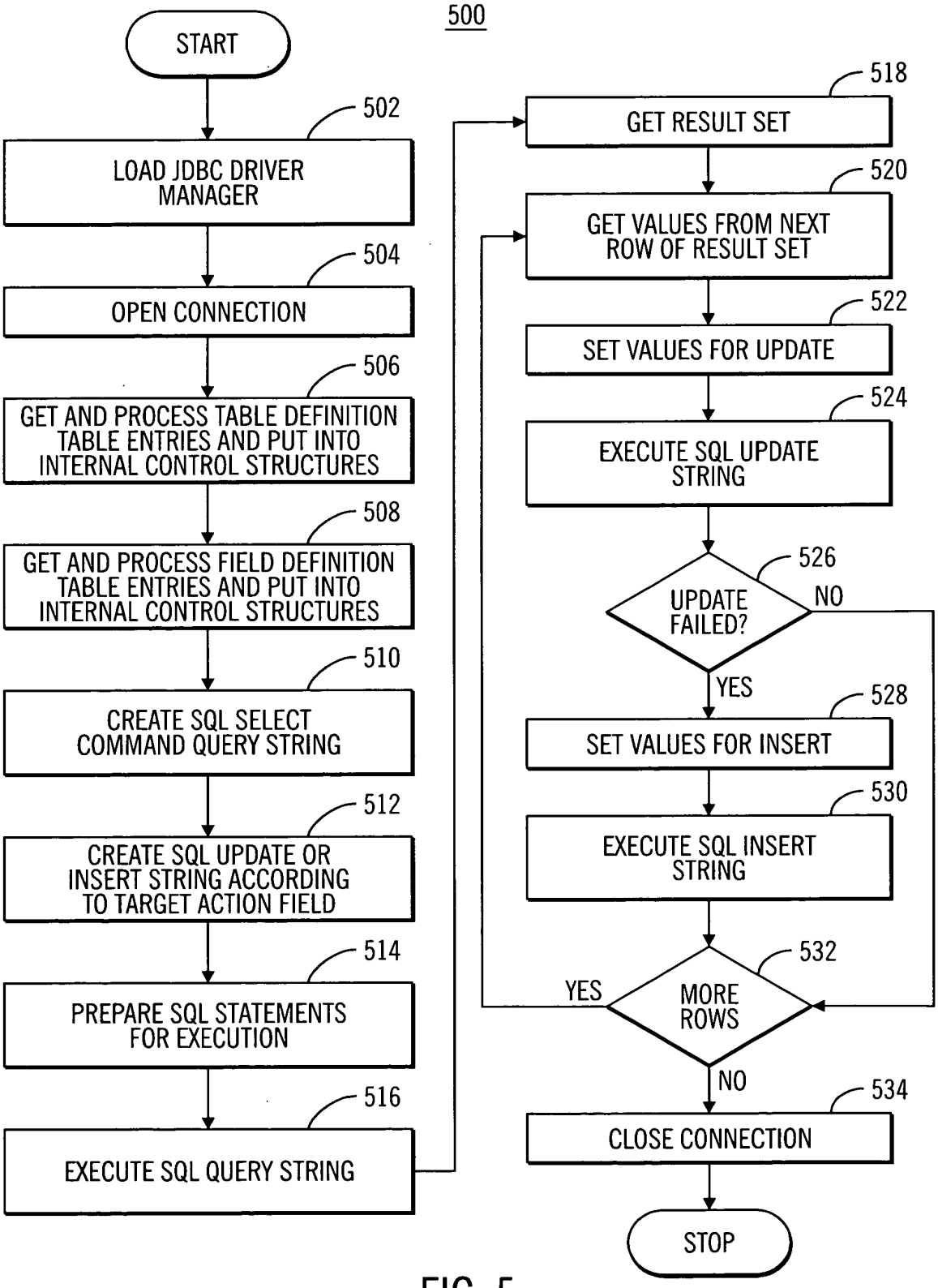

500
START
502
LOAD JDBC DRIVER MANAGER
504
OPEN CONNECTION
506
GET AND PROCESS TABLE DEFINITION TABLE ENTRIES AND PUT INTO INTERNAL CONTROL STRUCTURES
508
GET AND PROCESS FIELD DEFINITION TABLE ENTRIES AND PUT INTO INTERNAL CONTROL STRUCTURES
510
CREATE SQL SELECT COMMAND QUERY STRING
512
CREATE SQL UPDATE OR INSERT STRING ACCORDING TO TARGET ACTION FIELD
514
PREPARE SQL STATEMENTS FOR EXECUTION
516
EXECUTE SQL QUERY STRING
518
GET RESULT SET
520
GET VALUES FROM NEXT ROW OF RESULT SET
522
SET VALUES FOR UPDATE
524
EXECUTE SQL UPDATE STRING
526
UPDATE FAILED?
NO
YES
528
SET VALUES FOR INSERT
530
EXECUTE SQL INSERT STRING
532
YES
MORE ROWS
NO
534
CLOSE CONNECTION
STOP

FIG. 5

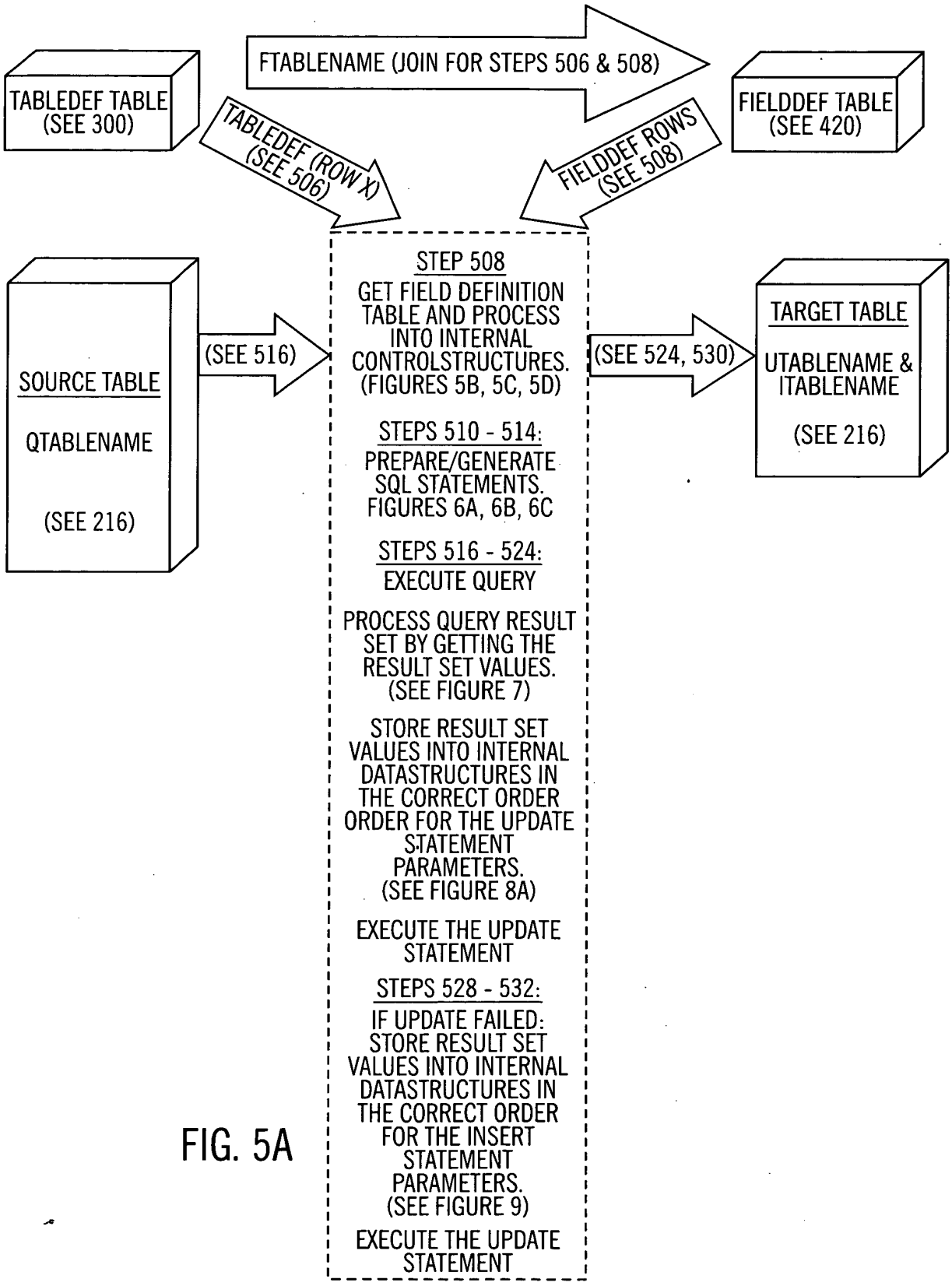

FTABLENAME (JOIN FOR STEPS 506 & 508)
TABLEDEF TABLE (SEE 300)
FIELDDEF TABLE (SEE 420)
TABLEDEF (ROW X) (SEE 506)
FIELDDEF ROWS (SEE 508)
SOURCE TABLE
QTABLENAME
(SEE 216)
(SEE 516)
STEP 508
GET FIELD DEFINITION TABLE AND PROCESS INTO INTERNAL CONTROLSTRUCTURES. (FIGURES 5B, 5C, 5D)
STEPS 510 - 514:
PREPARE/GENERATE SQL STATEMENTS. FIGURES 6A, 6B, 6C
STEPS 516 - 524:
EXECUTE QUERY
PROCESS QUERY RESULT SET BY GETTING THE RESULT SET VALUES. (SEE FIGURE 7)
STORE RESULT SET VALUES INTO INTERNAL DATASTRUCTURES IN THE CORRECT ORDER ORDER FOR THE UPDATE STATEMENT PARAMETERS. (SEE FIGURE 8A)
EXECUTE THE UPDATE STATEMENT
STEPS 528 - 532:
IF UPDATE FAILED: STORE RESULT SET VALUES INTO INTERNAL DATASTRUCTURES IN THE CORRECT ORDER FOR THE INSERT STATEMENT PARAMETERS. (SEE FIGURE 9)
EXECUTE THE UPDATE STATEMENT
(SEE 524, 530)
TARGET TABLE
UTABLENAME & ITABLENAME
(SEE 216)

FIG. 5A

550

```
Select SCLAUSE, JTYPE, SFLDSEQ FROM UnitStatTab WHERE SCLAUSE is
NOT NULL ORDER BY SFLDSEQ

i=1
While not end of resultset
{
// Get values from next row of resultset
 SClause[i] = GetString [1]          // get SCLAUSE
 SJType[i] = GetString [2]           // get JTYPE      551
 SFldSeq[i] = GetInt [3]             // get SFldSeq
i++
}
Ssize = i--:
```

FIG. 5B1

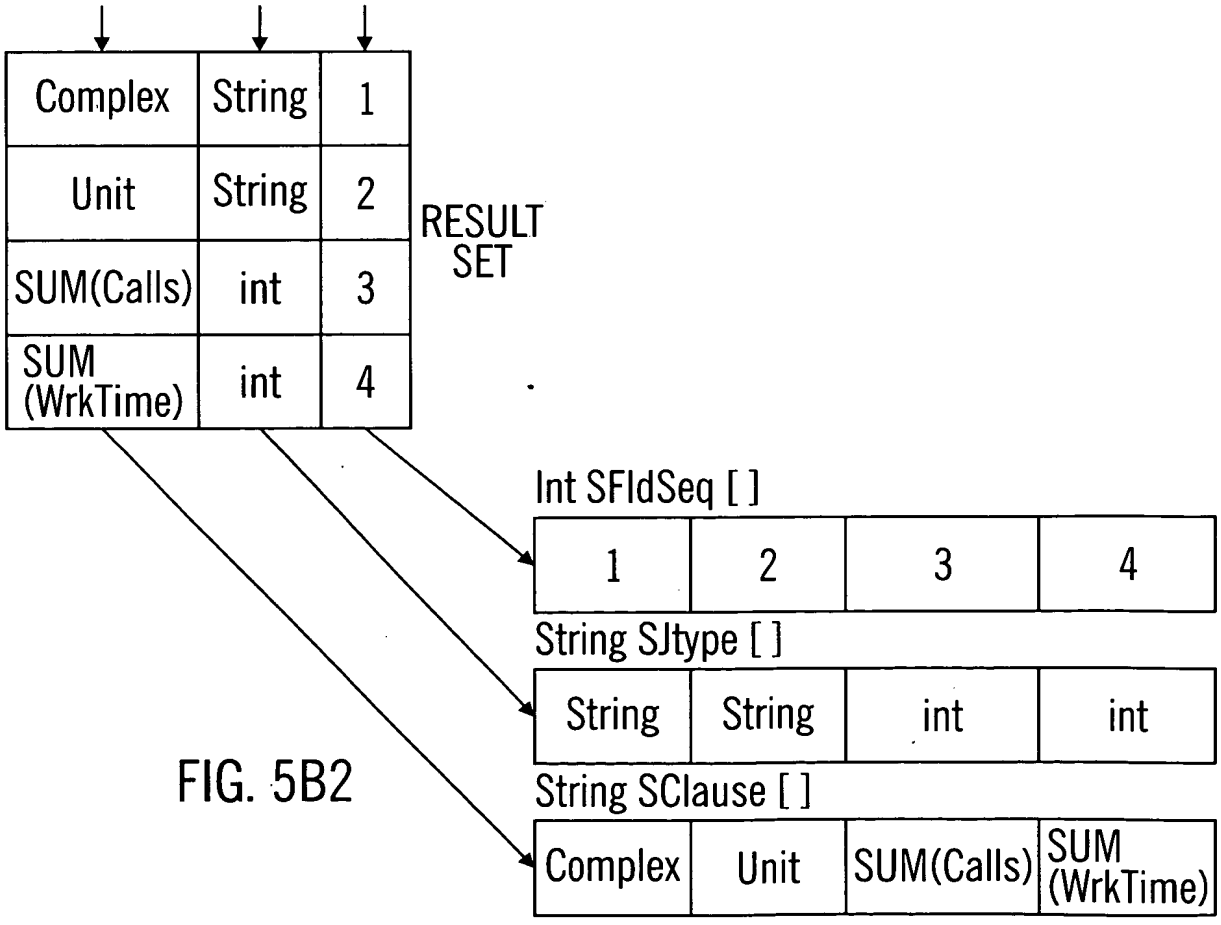


FIG. 5B2

```
                                                    556
Select GCLAUSE, SFLDSEQ FROM UnitStatTab WHERE GCLAUSE is
                                        NOT NULL ORDER BY SFLDSEQ

i=1
While not end of resultset
{
// Get values from next row of resultset
GClause[i] = GetString [i]              // get GCLAUSE      557
i++
}
Gsize = i--:
```

FIG. 5B3

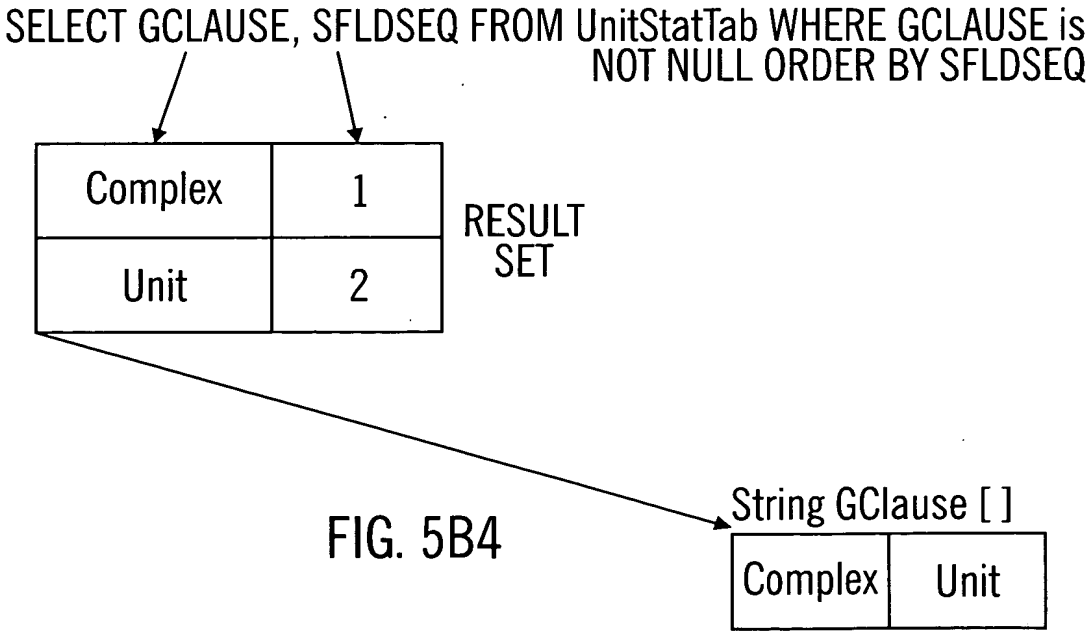


FIG. 5B4

565

```
SELECT UCLAUSE, JTYPE, SFLDSEQ, TFLDSEQ FROM UnitStatTab WHERE
                                UCLAUSE is NOT NULL ORDER BY TFLDSEQ
    i=1
    While not end of resultset
    {
    // Get values from next row of resultset
     UClause[i] = GetString [1]          // get UCLAUSE
     UJType[i] = GetString [2]           // get JTYPE        566
     USFldSeq[i] = GetInt [3]            // get SFldSeq
    UFldseq[i] = i                       // reset sequence
    i++
    }
    Usize = i--:
```

FIG. 5C1

SELECT UCLAUSE, JTYPE, SFLDSEQ, TFLDSEQ FROM UnitStatTab WHERE UCLAUSE is NOT NULL ORDER BY TFLDSEQ

Calls=?	int	3	3
WrkTime=?	int	4	4
CumCalls= CumCalls+?	int	3	5
CumWrkTime= CumWrkTime+?	int	4	6

RESULT SET

Int UFldSeq [ ]

1	2	3	4

Int USFldSeq [ ]

3	4	3	4

String UJtype [ ]

int	int	int	int

String UClause [ ]

Calls=?	WrkTime=?	CumCalls= CumCalls+?	CumWrkTime= CumWrkTime+?

FIG. 5C2

574

```
SELECT WCLAUSE, SFLDSEQ, TFLDSEQ FROM UnitStatTab WHERE
                    WCLAUSE is NOT NULL ORDER BY TFLDSEQ
  i=1
  While not end of resultset
  {
  // Get values from next row of resultset
   WClause[i] = GetString [1]          // get WCLAUSE
   WJType[i] = GetString [2]           // get JTYPE        575
   WSFldSeq[i] = GetInt [3]            // get SFldSeq
   WFldseq[i] = Usize + i              // reset sequence
  i++
  }
  Wsize = i--:
```

FIG. 5C3

SELECT WCLAUSE, SFLDSEQ, TFLDSEQ FROM UnitStatTab WHERE WCLAUSE is NOT NULL ORDER BY TFLDSEQ

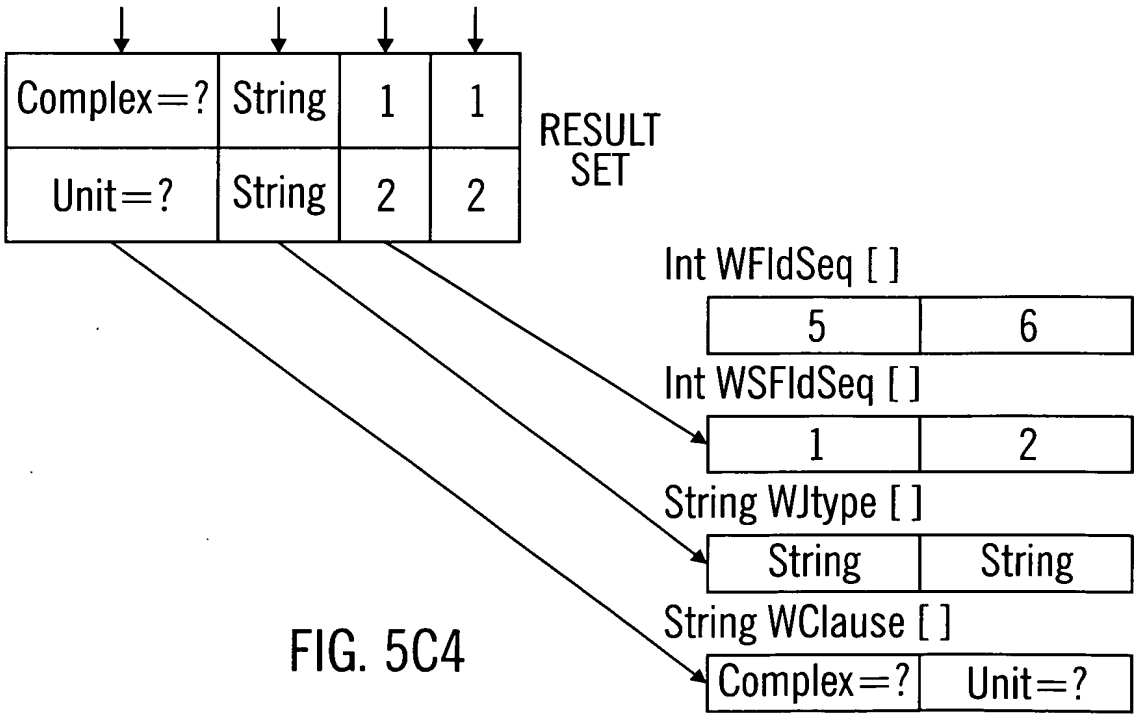


FIG. 5C4

```
                                                        580
SELECT ICLAUSE, JTYPE, SFLDSEQ, TFLDSEQ FROM UnitStatTab WHERE
                                    ICLAUSE is NOT NULL ORDER BY TFLDSEQ
    i=1
    While not end of resultset
    {
    // Get values from next row of resultset
     IClause[i] = GetString [1]          // get ICLAUSE
     IJType[i] = GetString [2]          // get JTYPE          581
     ISFldSeq[i] = GetInt [3]           // get SFldSeq
    IFldseq[i] = i                      // reset sequence
    i++
    }
     Isize = i--:
```

FIG. 5D1

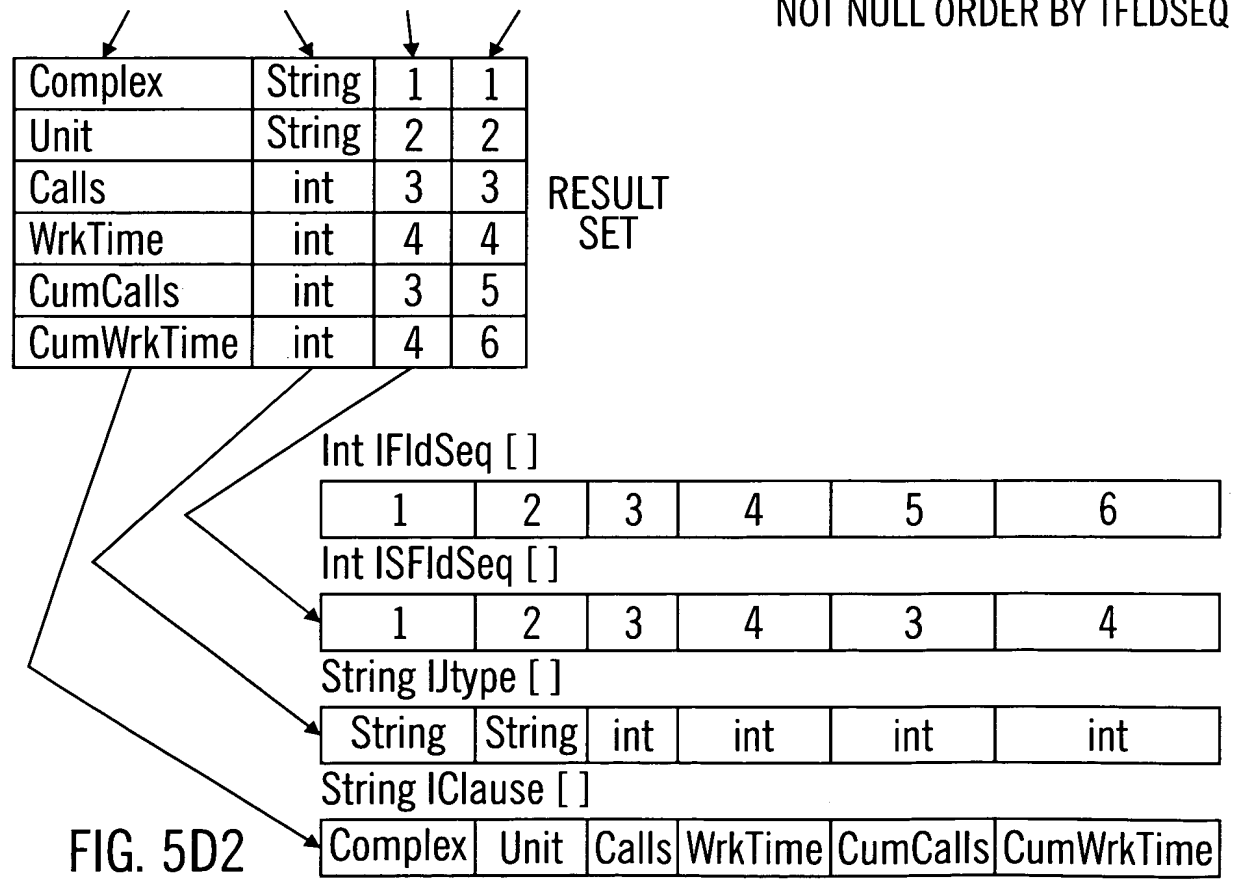


FIG. 5D2

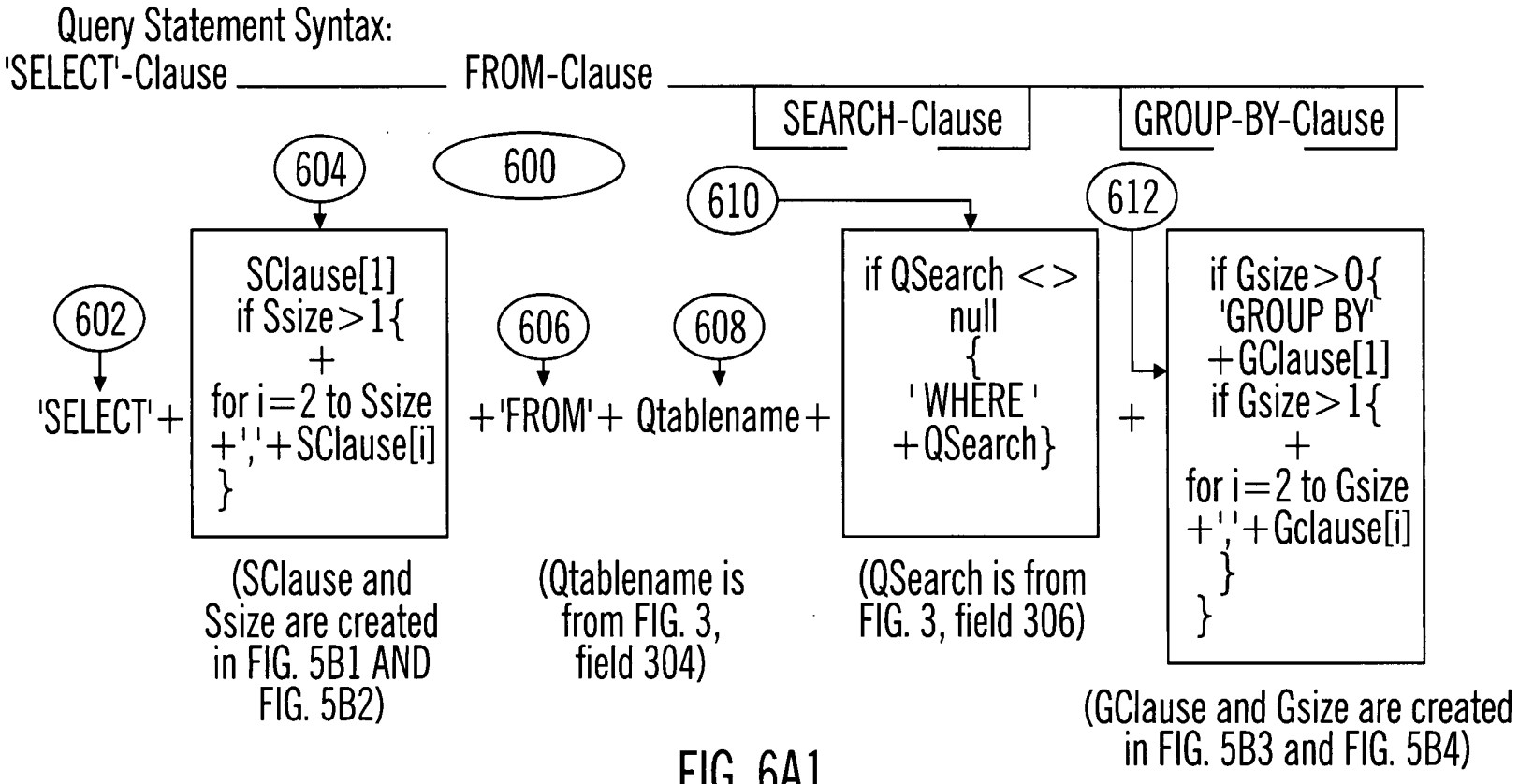


FIG. 6A1

Select Statement String which was built from 600 above.

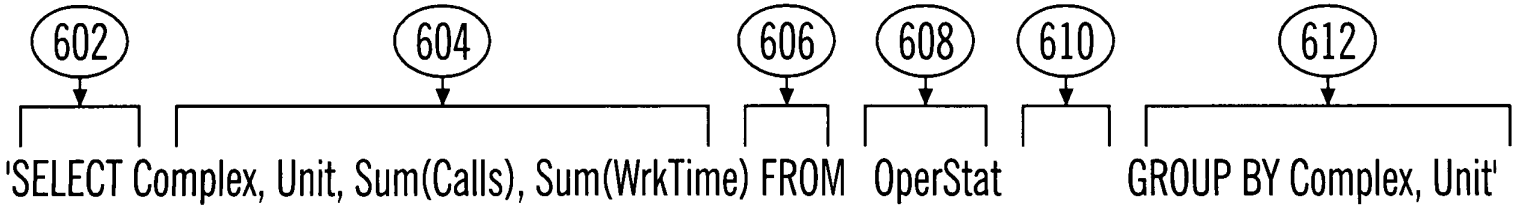


FIG. 6A2

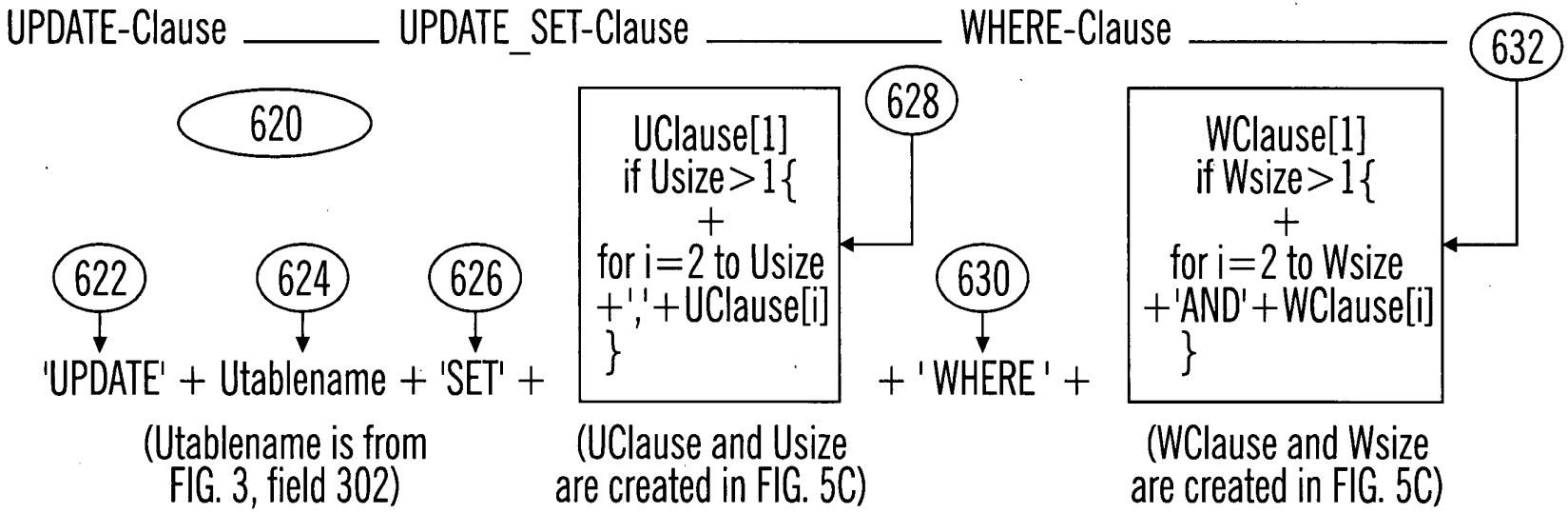


FIG. 6B1

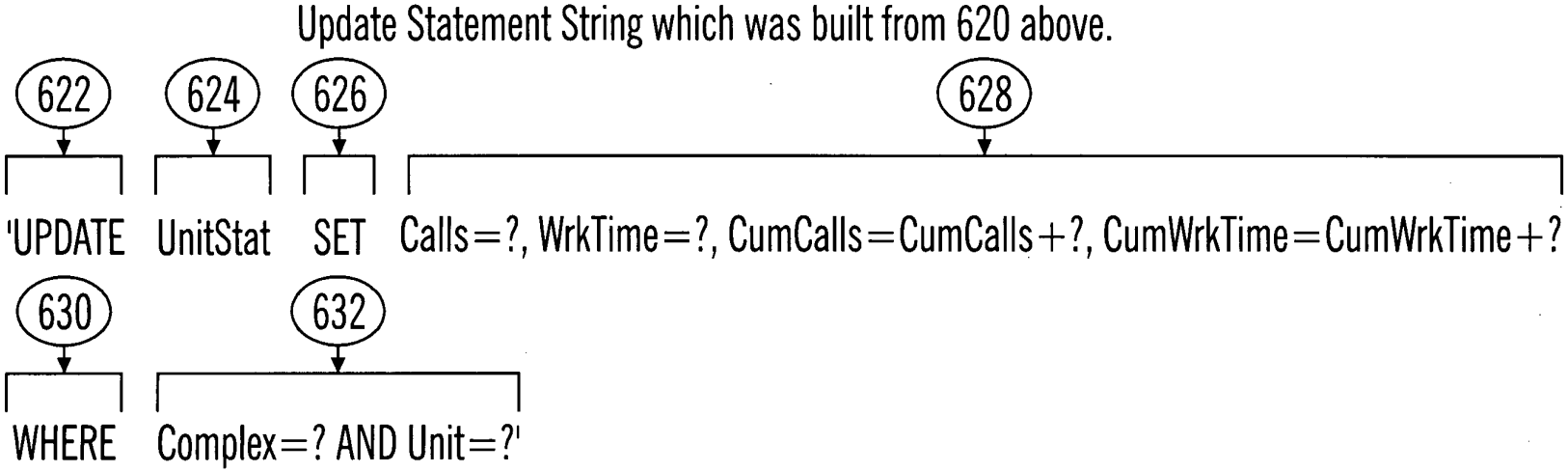


FIG. 6B2

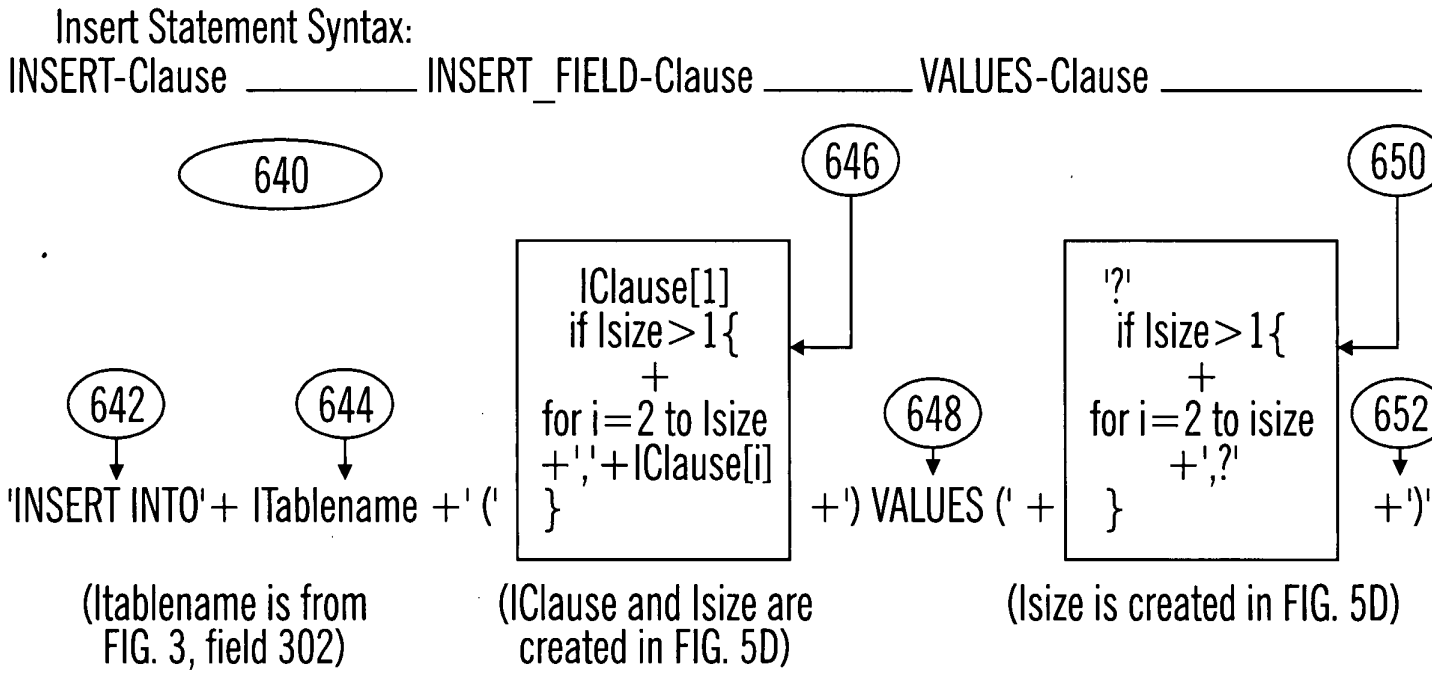


FIG. 6C1

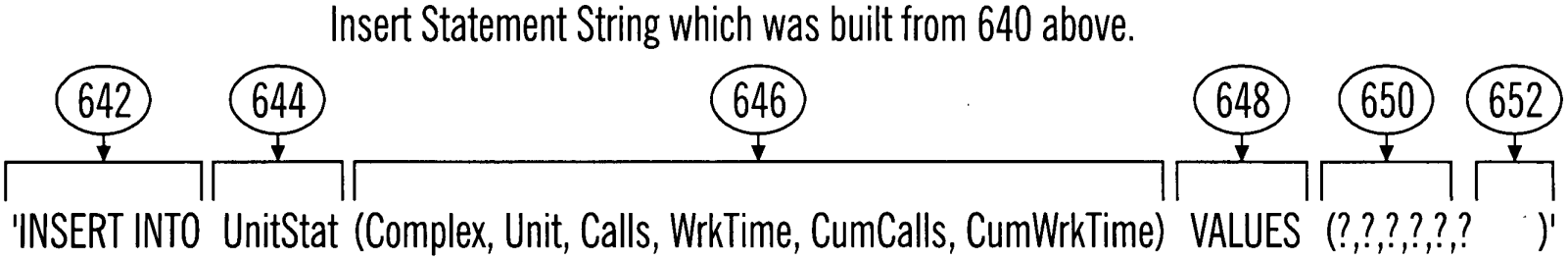


FIG. 6C2

550

```
'SELECT Complex, Unit, Sum(Calls), Sum(WrkTime) FROM OperStat
                                        GROUP BY Complex, Unit'

  For i=1 to Ssize
  {
    j=SFIdSeq[i]
    if SJType[i]='String'
      StringRS[j]=GetString[i]
  else
      IntRS[j]=Getint[i]
  }
```

FIG. 7A

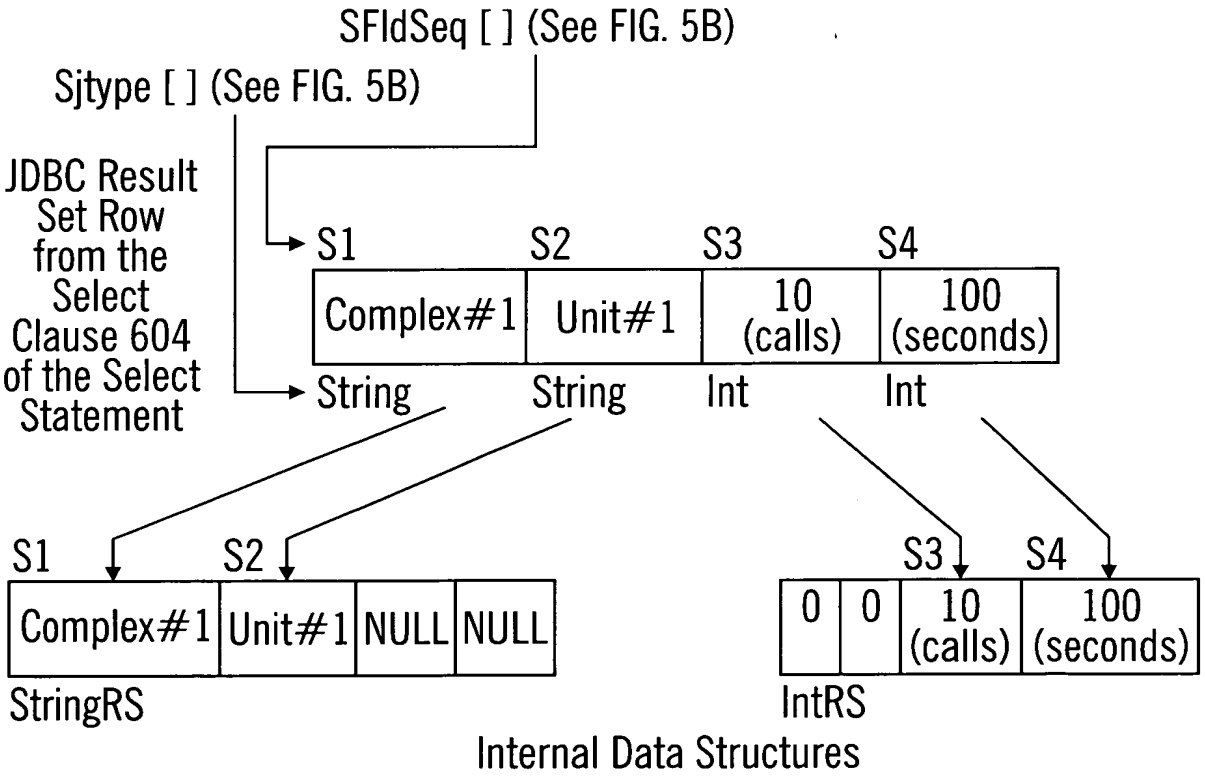


FIG. 7B

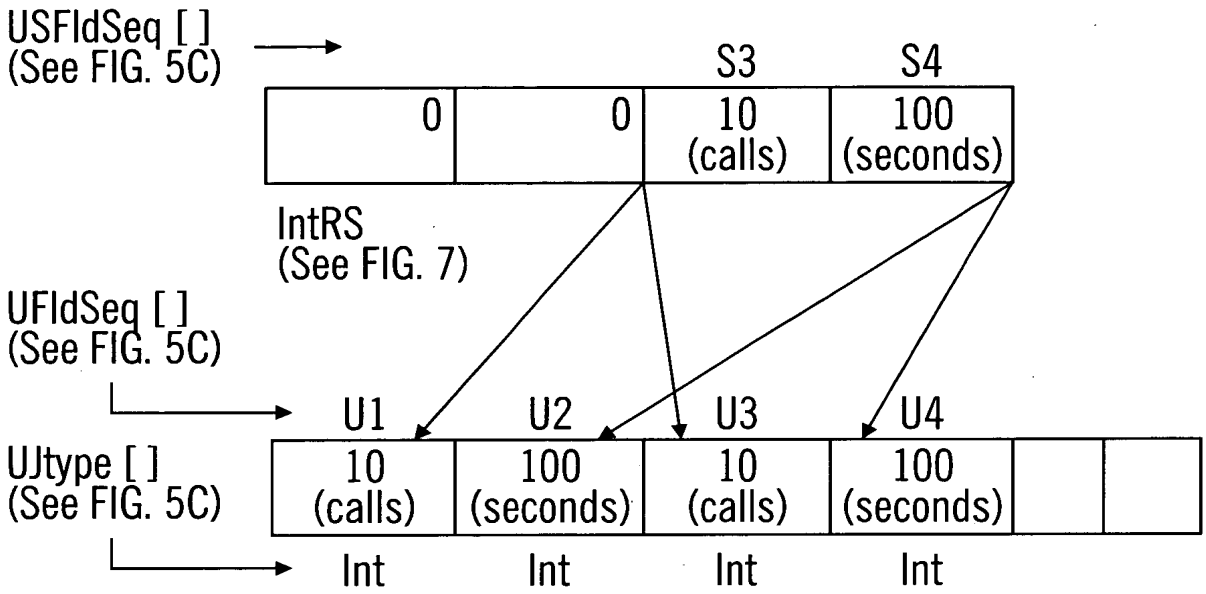
```
'UPDATE UnitStat SET Calls=?, WrkTime=?, CumCalls=CumCalls+?,
                       U1       U2                    U3
                                          CumWrkTime=CumWrkTime+?
                                                        U4
 WHERE Complex=? AND Unit=?'
              U5          U6
```

```
For i=1 to Usize
{
  j=UFldSeq[i]
  k=USFldSeq[i]
  if UJType[i]='String'
    setString[j]=StringRS[k]
  else
    setint[j]=IntRS[k]
}
```

FIG. 8A1

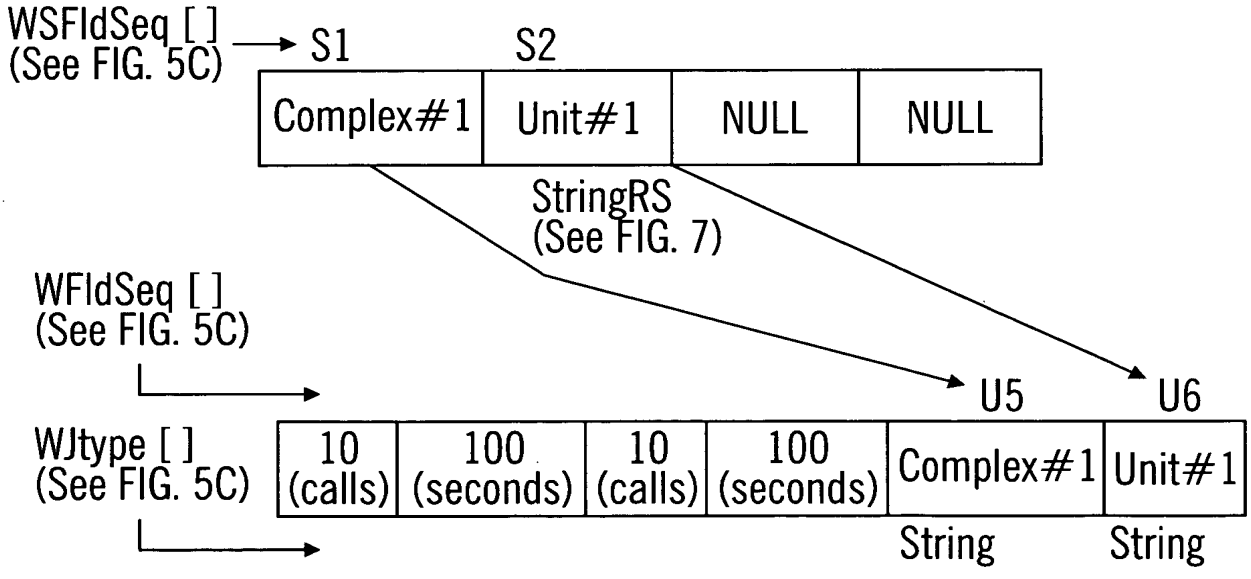
```
For i=1 to Wsize
{
  j=WFldSeq[i]
  k=WSFldSeq[i]
  if WJType[i]='String'
    setString[j]=StringRS[k]
  else
    setint[j]=IntRS[k]
}
```

FIG. 8A2



JDBC parameters for Update_Set Clause 628 of Update Statement

FIG. 8A3



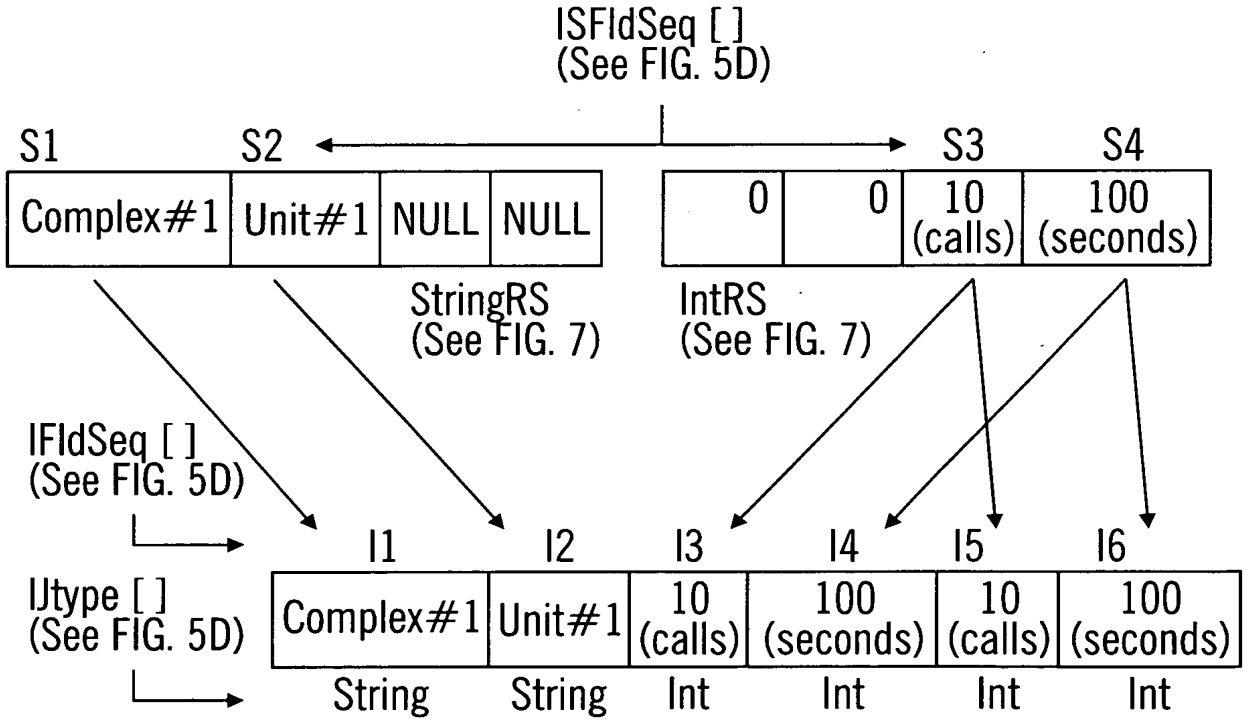
JDBC parameters for Where Clause 632 of Update Statement

FIG. 8A4

```
'INSERT INTO  UnitStat(Complex, Unit, Calls, WrkTime, CumCalls,
CumWrkTime) VALUES (?,?,?,?,?,?)'
                              I1,I2,I3,I4,I5,I6

For i=1 to Isize
{
  j=IFldSeq[i]
  k=ISFldSeq[i]
  if IJType[i]='String'
    setString[j]=StringRS[k]
 else
   setint[j]=IntRS[k]
}
```

FIG. 9A



JDBC parameters for Values Clause 650 of Insert Statement

FIG. 9B